FUDA EMPIRICAL EVALUATION
Development Experience Questionnaire (Used Templates)

NOTE: This is a fillable PDF file. If you are not using a PDF writer, please make sure to print this file to a PS or PDF file to not lose the information.

Name: S16 Date: 17 August 2008

Concept: □ JFace – Context Menu □ JFace – Content Assist
□ Eclipse – Navigate ☑ Eclipse – Table Viewer

Q.1: Were you able to implement the concept successfully? ☑ Yes □ No

Q.2: How much time did you spend on the concept's implementation? 80 minutes

Q.3: If not successful to implement the concept, what was the main reason in your opinion?

□ Lack of experience.
□ Not a useful template.
□ Not useful sample applications.
□ Complexity of the concept.
□ Other. Please specify: ______

Q.4: Did you refer to the example applications' source code to implement the concept?

□ No. None of them. □ Yes. One of them. Please specify: ______ ☑ Yes. Both of them.

Q.4.1: If yes, for what program statements and what kind of information?

- For finding the contents of the getElements() method on FRL_03.
- For finding out what should be passed to setInput() method as a parameter on FRL_20

Q.5: Overall, did you find the templates useful? If yes, in what way? If not, why?

It was definitely useful. The source code was there and I did not need to think about which methods and classes should be implemented. The template had even provided the required contents for most of the methods, making my work much easier. It had also provided the type of variables and parameters and had explicitly declared the required variables, so I did not have to know any extra information about those methods to be able to use them.

Q.6: Do you think that the format and structure of the templates are OK? If not, what are the main issues?

I like the format. In my opinion, being close to Java language is one of the best characteristics of the template format. It reduces the amount of work to be done by developers extensively.

Q.7: What kinds of information do you think are missing in the templates?

It will be better if the templates point out those parts of the template which developers should change and add some extra statements to them. For example in the TableViewer template there are three methods for AppStructuredContentProvider class. It seems that 2 of those methods do not need any extra statements added to them unlike the getElements() method which needs some implementation by developer. In the first look, all of those 3 methods look alike in the template. Hence it is a little bit confusing for the developer to know if he is required to add anything to the method or not, just using the template itself.

Q.8: Overall, in the range of 1-5, how do you rank the provided template in terms of usefulness to implement the concept?

□ 1 = Not Useful   □ 2   □ 3   ☑ 4   □ 5 = Excellent

Q.9: Do you have any additional comments on this experiment?

I believe that some factors should be considered before any further interpretation of the experience. First of all this was my first time trying to develop an Eclipse plug-in. If I am supposed to implement another plug-in it will definitely require much less time. Secondly this was my first experience with the template. It takes time to get used to a new tool even if the tool itself is perfect. I had a hard time to find out how I should add a cell to the table in the getElements() method. The template had nothing implemented for this method.

Additional Space: